



REAALIAIKAISEN STRATEGIAPELIN KEHITTÄMINEN UNITY-PELINKEHITYSTYÖKALULLA

Case Rescue 2: Everyday Heroes

Juho Lyytikäinen

Opinnäytetyö
Joulukuu 2015
Tietojenkäsittely
Ohjelmistokehitys



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittely

LYYTIKÄINEN, JUHO:

Reaaliaikaisen strategiapelin kehittäminen Unity-pelinkehitystyökalulla

Case: Rescue 2: Everyday Heroes

Opinnäytetyö 45 sivua
Joulukuu 2015

Opinnäytetyössä käsitellään Rescue 2: Everyday Heroes-pelin kehitysprojektia. Rescue 2 on reaaliaikainen pelastusaiheinen strategiapeli, joka on kehitetty käyttäen Unity-pelinkehitystyökalua. Opinnäytetyön toimeksiantajana toimii Fragment Production Oy-niminen peliyritys.

Opinnäytetyön tavoitteena on yrityksen toiminnan, prosessien sekä kehitettävien pelien laadun parantaminen. Työn tarkoituksena on kuvata kehitysprosesseja ja arvoida projektin onnistumista, käytettyjä menetelmiä sekä työkaluja. Menetelmiin kuuluu projektin toteutusten dokumentointi, arviointi ja kehitysehdotukset.

Projekti kohtasi useita ongelmia tuotannon aikana. Pelin laajuus, vaihtuva henkilöstö sekä ajoittain heikko kommunikaatio tekivät tuotannosta haastavaa. Useat yhtäaikaiset projektit yrityksessä lisäsivät haasteita projektin tuotannossa.

Unity oli luonnollinen valinta kehitystyökaluksi jatko-osalle, koska myös pelisarjan ensimmäinen osa oli toteutettu Unityllä. Unity osoittautui muutenkin hyväksi valinnaksi pelin ominaisuuksien puolesta. Tulevissa projekteissa tulee kuitenkin harkita Unityn käyttöä, sillä muut pelinkehitystyökalut voivat soveltua paremmin tuotantoon. Projektin kulun suunnittelua parantamalla ja projektinhallintamenetelmiä kehittämällä voidaan saavuttaa paremmat tulokset tulevissa projekteissa.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Business Information Systems

LYYTIKÄINEN, JUHO:

Development of a Real-Time Strategy Game Using Unity Game Development Platform
Case Rescue 2: Everyday Heroes

Bachelor's thesis 45 pages
December 2015

This thesis covers the development project of Rescue 2: Everyday Heroes, a real-time strategy game themed around firefighting and first aid. The game was developed using the Unity game development tool. The thesis was made for a game development company called Fragment Production Ltd.

The objective of the thesis was to improve the quality of the games developed by the company, and to improve the processes used for production. The thesis describes the development of the game by evaluating the success of the project and the tools and procedures used. The methods include documentation and evaluation of implementations and suggestions for improving the processes in the company.

The project faced multiple problems during development, ranging from those posed by the scale of the game to major changes in the human resources.

Unity was chosen as the tool for development, as the game under development was a sequel to a previous game developed by the company. Unity performed well in the context of implementing the required features for the game. In upcoming projects it would be advisable to plan the production more carefully, evaluate the usage of Unity in terms of the product and revise the methods for project management.

Key words: game development, Unity, project management

SISÄLLYS

1	JOHDANTO.....	8
2	TAUSTAA	10
2.1	Tavoite ja tarkoitus	10
2.2	Projektin taustaa.....	11
2.3	Unity-pelinkehitystyökalu	11
3	RESCUE 2: EVERYDAY HEROES – PROJEKTI.....	13
3.1	Pelin kuvaus	13
3.1.1	Asemapelitila.....	13
3.1.2	Karttapelitila.....	14
3.1.3	Tehtäväpelitila.....	15
3.1.4	Tutoriaalipelitila.....	17
3.2	Projektin läpivienti.....	17
3.2.1	Alkutuotanto.....	18
3.2.2	Ensimmäinen pelattava versio.....	18
3.2.3	Alpha versio	19
3.2.4	Beta versio.....	19
3.2.5	Gold Master versio	20
3.2.6	Julkaisu.....	21
3.2.7	Jälkituotanto	22
4	PROJEKTIN TOTEUTUS	23
4.1	Suunnittelu	23
4.2	Projektin kulku.....	24
4.2.1	Projektinhallinta	24
4.2.2	Versionhallinta	25
4.2.3	Testaus ja virheiden seuranta	26
4.3	Keskeisten ominaisuuksien toteutus	27
4.3.1	Tulisimulaatio	28
4.3.2	Komentojärjestelmä	28
4.3.3	Yksiköiden hallinta	29
4.3.4	Pelin eteneminen	30
4.4	Unity-laajennukset	30
4.4.1	Localization Package	31
4.4.2	NGUI (Next-Gen UI)	32
4.4.3	HOTween	33
4.4.4	Ludosity Steamworks.....	34
4.4.5	MasterAudio.....	35

4.4.6 VPaint.....	36
4.4.7 A* Pathfinding Project.....	37
4.4.8 Decal System.....	38
4.5 Ohjelmointiympäristö	39
4.6 Audio toteutus.....	40
4.7 Graafinen toteutus.....	41
4.8 Markkinointi materiaalin luonti	41
5 POHDINTA.....	43
LÄHTEET.....	45

LYHENTEET JA TERMIT

Asemapelitila	Pelin tila, missä pelaaja hallinnoi valitsemaansa aseman eri yksiköitä sekä kalustoa.
Atlas	Kuva joka, sisältää useita yksittäisiä kuvia, jota käyttämällä voidaan vähentää piirtokäskeyjen määrää.
CryEngine	CryTek-yrityksen kehittämä pelinkehitystyökalu.
DirectX	Microsoftin kehittämä rajapinta multimedia toimintojen suorittamiseen erityisesti pelin kehityksessä.
GDD	Peli suunnitteludokumentti, joka sisältää pelin ominaisuudet ja niiden olennaiset yksityiskohdat (engl. Game Design Document).
Karttapelitila	Pelin tila, missä pelaaja hallinnoi kartalle olemassa olevia yksiköitä (ajoneuvoja) lähettäen niitä eri tehtävillä tai asemille.
Lokalisointi	Pelin kielen kääntäminen eri kielille liittyen pääasiallisesti pelin tekstin kääntämisen eri kielelle. Pelin kuvat sekä äänet voidaan myös tarvittaessa kääntää eri kielille.
Mercurial	Matt Mackall'in kehittämä hajautettu versionhallintajärjestelmä, joka tukee yleisimpiä käyttöjärjestelmiä.
Porttaus	Pelin muokkaaminen siten, että peli toimii eri alustoilla kuin vain sillä millä tai mille se on kehitetty.
Scrum	Projektinhallinnan viitekehys, jota käytetään yleisemmin ketterään ohjelmistokehitykseen, jota voidaan soveltaa yleisesti projektinhallinnassa.
Steam	Valve Corporation-yrityksen (yleisemmin tunnettu Valve) videopelien digitaaliseen jakeluun erikoistunut yritys. Pelien lisäksi Steam jakelee myös eri tietokoneohjelmistoa.
Sävytin	Tietokoneohjelma, jolla voidaan laskea piirrettävän pinnan.
Tehtäväpelitila	Pelin tila, missä pelaaja kontrolloi yksittäisiä yksiköitä vastaan eri hälytys tilanteisiin. Tilanteisiin liittyy eri tavoitteita, mitä pelaajan on suoritettava tehtävän suorittamiseen.
Tekstuuri	3D-mallin pinnalle piirrettävä kaksiulotteinen kuva.

Tutoriaalipelitila	Tehtäväpelitilan alitila, jossa pelaajaa opastetaan eri pelin ominaisuuksiin suorittamalla tutoriaali-tehtävän tavoitteita ohjeiden kera.
Tween	Animaation muoto, missä muutetaan arvoja kahden luvun välillä ajan kuluessa.
UDK	Epic Games-yrityksen kehittämä ilmainen versio Unreal Engine-pelimoottorista.
Unity	Unity Technologies-yrityksen kehittämä pelinkehitystyökalu, mikä sisältää pelimoottorin sekä graafisen käyttöliittymän pelien luontia varten. Unityä on mahdollista käyttää eri alustoilla sekä luoda pelejä eri alustoille.
Unity-laajennus	Kolmannen osapuolen kehittämä laajennus Unity-pelinkehitystyökalua varten, mikä on joko vapaasti saatavilla tai jake-lussa Unityn Asset Store-palvelusta.
Versionhallinta	Tekniikka jolla pidetään kirjaa eri tiedostoihin tehdyistä muutoksista ajan kuluessa ja säilötään vanhat versiot.
Välietappi	Projektiin suunniteltu etappi, missä aiemmin sovitut asiat on toteutettu helpottaen projektin kehityksen seuranta.

1 JOHDANTO

Opinnäytetyö käsittelee pelinkehitystä Unity-pelinkehitystyökalulla arvioiden toimeksiantajan tuottamaa tietokonepeliä, minkä avulla yritys pyrkii parantamaan toimintaa tulevien peliprojektien tuotannossa sekä antamaan kattavan kuvan käytetystä pelinkehitystyökalusta työntekijöiden orientoitumista ja tuotannon prosesseja varten. Opinnäytetyön toimeksiantajana toimii Tamperelainen yritys nimeltä Fragment Production Oy, joka kehittää strategia- & simulaatio pelejä eri alustoille.

Fragment Production Oy on Tamperelainen vuonna 2012 perustettu yritys, joka kehittää tietokone- ja mobiilipelejä. Yritys työllistää yli 20 työntekijää ja keskittyy strategia & simulaatio genren pelien kehittämiseen eri alustoille. Vuoteen 2015 kesään mennessä yritys on kehittänyt neljä eri peliä usealle eri alustalle, joissa julkaisijana on toiminut saksalainen pelijulkaisija rondomedia GmbH.

Tietokonepeli alustoihin kuuluu Microsoft Windows sekä Mac OS X ja mobiili alustoihin kuuluu Windows Phone 8, Android sekä iOS. Julkaisija projektien ohessa yritys myös kehittää itse julkaistavia projekteja. Yrityksen henkilöstöön kuuluu usean eri alueen osajia kuten pelisuunnittelijoita, ohjelmoijia, graafikkoja ja tuottajia.

Unity (yleisemmin tunnettuna Unity3D) on Unity Technologies yrityksen julkaisema pelinkehitystyökalu, joka sisältää pelimoottorin sekä kehitysympäristön. Unityn ensimmäinen versio julkaistiin vuonna 2005. Aluksi Unity oli vain saatavilla OS X alustalle, mutta myöhemmissä versioissa tulivat useat eri alustat tuetuiksi. Nykyinen versio 5.2. tukee mobiili, pöytätietokone, selain sekä konsoli alustoja. Tuettuihin alustoihin kuuluu Apple iOS, Android, Windows Phone 8, Blackberry 10, Tizen, Windows, Windows Store Apps, MAC OS X, Linux/Steam OS, Oculus Rift, Gear VR, Microsoft Hololens, PlayStation 3, Playstation 4, Playstation Vita, XBOX One, XBOX 360, Wii U sekä selaimet (Unity 2015a).

Unitystä on saatavilla ilmainen sekä maksullinen versio (Unity 2015d), missä ominaisuudet poikkeavat hieman. Pelimoottorin lähdekoodi on saatavilla sovittavaan hintaan yrityksille. Vuonna 2015 julkaistun Unity 5-version myötä muuttuivat lisenssiehdot niin, että ilmainen versio korvautui henkilökohtaisella versiolla, missä ominaisuudet vastaavat

lähes kokonaan maksullista versiota. Henkilökohtainen versio soveltuu pienille pelistudioille sekä yksityisille käyttäjille.

Toteutetussa projektissa käytettiin Unityn maksullista versiota, versio numerolla 4.5.3. Projektin kuluessa versiota päivitettiin muutamaan kertaan, mutta huomioitavaa oli että päivitysten myötä saattoivat jotkut pelin ominaisuudet vioittua. Merkittävänä vaihtoehtoina Unitylle olisivat esimerkiksi yrityksen Epic Gamesin julkaisema UDK tai CryTek:in julkaisema CryEngine. Projektin tilanteesta ja yrityksen historiasta johtuen oli Unity parempi vaihtoehto projektin onnistumisen takaamiseen. Projektin alun aikana myös lisenssiehdot olivat hyvin erilaiset eri työkalujen välillä, sillä Unity ei vaatinut rojalteja pelin myynnistä.

2 TAUSTAA

Opinnäytetyö käsittelee Rescue 2: Everyday Heroes tietokonepelin kehitysprojektia, jonka kehitti Fragment Production Oy ja minkä julkaisijana toimi saksalainen pelijulkaisija rondomedia GmbH.

Projektin kulku voidaan jakaa eri tuotannon vaiheisiin, mihin kuuluu projektin esituotanto, tuotanto ja jälkituotanto. Projektin esituotanto alkoi vuonna 2013 kesällä ja jatkui suoraan tuotantoon esituotannon loputtua. Tuotanto loppuu tyypillisesti pelin julkaisuun, mistä projekti voi jatkua jälkituotantoon tilanteesta riippuen. Projektin tuotanto loppui kesällä vuonna 2015 pelin julkaisuun, minkä jälkeen projektissa jatkettiin jälkituotantoon.

Opinnäytetyön tekijä toimi projektissa projektipäällikkönä sekä peliohjelmoijana. Projektin henkilöstöön kuului projektin ajan saatossa useita eri osa-alueen tekijöitä, mutta valtaosan ajasta henkilöstöön kuului: johtava graafikko, johtava peliohjelmoija, pelisuunnittelija, kenttäsuunnittelija, peliohjelmoijia sekä graafikkoja.

2.1 Tavoite ja tarkoitus

Opinnäytetyön tavoitteena on parantaa yrityksen toimintaa, prosesseja sekä kehitettävien pelien laatua. Yrityksen perustusvaiheessa yrityksen tuotanto keskittyi yhteen pelituotantoon, mutta yrityksen kasvaessa tuotantoon tuli useita eri projekteja sekä kehitystiimejä. Tarkoituksena on kuvata Rescue 2: Everyday Heroes tietokonepelin kehitysprosessia arvioiden projektin onnistumista, käytettyjä menetelmiä sekä työkaluja.

Opinnäytetyö menetelmiin kuuluu Rescue 2: Everyday Heroes pelikehitys projektin toteutusten dokumentointi, arviointi ja kehitysehdotukset. Opinnäytetyössä käydään läpi projektin keskeisimmät ominaisuudet ja toteutukset, minkä johdosta voidaan arvioida käytettyjä prosesseja tuleviin peliprojekteihin.

2.2 Projektin taustaa

Rescue 2: Everyday Heroes on jatko-osa pelille Rescue 2013: Everyday Heroes. Rescue 2 on pelastusaiheinen reaaliaikainen strategiapeli, missä pelaaja hallitsee pelastustoiminnan yksiköitä hätätilanteissa, esimerkiksi sammutus- tai pelastustehtävissä. Tehtävien lisäksi pelaajan toimintaan kuuluu paloasemien hallinta, jossa hallitaan pelin yksiköitä osaten uusia pelastusajoneuvoja tai palkaten uutta henkilöstöä. Pelaajan on myös mahdollista kouluttaa henkilöstöstä ammattitaitoisempia kohdistamalla heidän keräämä kokemus haluttuihin taitoihin. Tehtävien suorittamisen ja asemien hallitsemisen ohessa pelaaja on vastuussa yksiköiden lähettämisestä eri kohteisiin, kuten tehtäville ja eri asemille.

Pelikehitysprojekti Rescue 2:delle alkoi syksyllä 2013, mutta yrityksen tehdessä muita projekteja oli projektin henkilöstö niukka vuoden 2014 alkuun saakka. Henkilöstö puutteen lisäksi itse projektihenkilöstö muuttui lähes kokonaan seuraavan vuoden aikana aiheuttaen ongelmia projektin tuotannossa. Ongelmien ja resurssien johdosta oli pelisuunnittelua muokattava, jotta peli saataisiin tuotettua. Peli julkaistiin kesäkuussa 2015 Windows-alustalle ja hieman myöhemmin MAC OS X alustalle Steam-palveluun, jonka jälkeen pelin jälkituotanto alkoi. Jälkituotannossa pelin virheitä korjattiin, lisättiin uusia käännettyjä kieliä sekä pelin etenemistä muokattiin käyttäjäpalautteen mukaisesti.

Pelin julkaisijana toimii saksalainen pelijulkaisija yritys nimeltä rondomedia GmbH, joka on toiminut toimeksiantajan kanssa yhteistyössä aikaisemmissa projekteissa julkaisten esimerkiksi pelin aikaisemman osan. Julkaisija keskittyy simulaatio- ja strategiapeligenren pelien julkaisuun, minkä alle myös kehitetty peli kuuluu.

2.3 Unity-pelikehitystyökalu

Unity on Unity Technologies -yrityksen kehittämä pelikehitystyökalu. Unity koostuu graafisesta editorista sekä pelimoottorista, mikä tarjoaa mahdollisuuden nopeaan pelin kehitykseen. Ohjelmakoodin kirjoittamista varten Unity tarjoaa käytettäväksi MonoDevelop ohjelmointiympäristön, mutta kehityksessä voi käyttää muitakin ohjelmointiympäristöjä, joka voi johtaa joidenkin ominaisuuksien puuttumiseen.

Yrityksellä ollessa kokemusta kyseisestä pelikehitystyökalusta oli projektin kannalta helpointa käyttää sitä pelin kehittämisessä. Unityn vahvuuksiin kuuluu myös tuki sekä

helppo siirrettävyys eri alustoille. Unityn tarjoama Asset Store palvelu, missä käyttäjät voivat ostaa ja myydä Unity-liitännäisiä, voi olla suuri apu kehittäessä pelejä pienillä tiimeillä.

2.3.1 Muut pelinkehitystyökalut

Unityn lisäksi vartenotettavia vaihtoehtoja ovat CryTek:n CryEngine ja Epic Games'in UDK(Unreal Development Kit). Molemmat ovat pääasiallisesti suunniteltu ensimmäisen persoonan ammunta peleihin, mutta niillä on myös mahdollista luoda erityyppisiä pelejä.

Eri pelinkehitystyökalujen lisensointi ehdot poikkeavat huomattavasti. CryEnginen ja UDK:n lisensointiin (Unreal Engine, 2015) kuuluu rojalTIMaksut pelimoottorin käytöstä, kun taas Unity ei sisällä rojalTIMaksuja, vaan kertamaksun tai tilauspohjaisen maksun. Vuonna 2015 useat eri pelinkehitystyökalut muuttivat lisensointi ehtojaan tehden esimerkiksi työkaluista ilmaisia sisältäen vain rojalTIMaksut, tehden työkalujen käyttöönotosta edullisempaa.

3 RESCUE 2: EVERYDAY HEROES – PROJEKTI

3.1 Pelin kuvaus

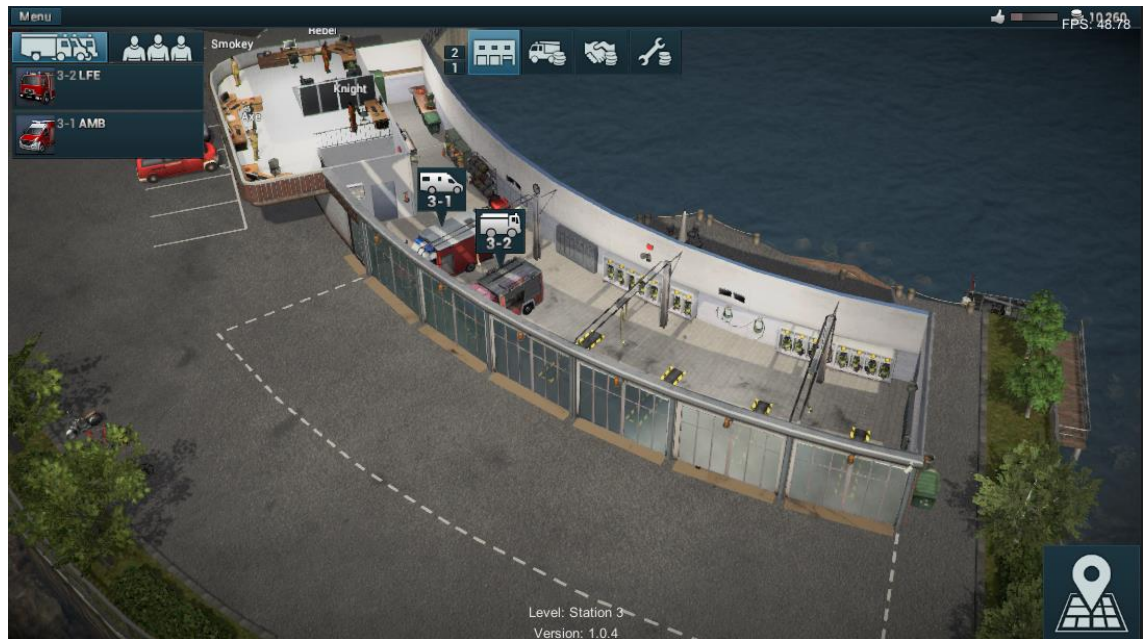
Rescue 2: Everyday Heroes on pelastusaiheinen reaaliaikainen strategiapeli, missä pelaaja hallitsee eri pelastusyksiköitä asemilla, kartalla sekä yksityiskohtaisilla tehtävillä. Peli jakautuu kolmeen eri päätilaan, joiden avulla pelaaja etenee pelissä suorittaen tehtäviä, avaten uusia ominaisuuksia sekä kouluttamalla henkilöstöä.

3.1.1 Asemapelitila

Pelimaailmaan liittyy paloasemia, minkä kautta pelaaja hallitsee pelin yksiköitä sekä työkaluja. Kaikkiaan paloasemia on kolme, mistä pelaajalle ensimmäinen avataan ilmaiseksi. Pelin edetessä pelaajalla on mahdollista ostaa uusia paloasemia, milloin pelaaja voi hallita suurempaa määrää yksiköitä sekä lähettää yksiköitä tehtäville aseman sijainnin mukaan saavuttaen tehokkaamman toiminnan kaupungissa. Asemalle ostetut ajoneuvot ja palkatut henkilöt voidaan nähdä kyseisellä asemalla, missä pelaaja voi muokata niiden ominaisuuksia.

Ajoneuvoille määrätty aseman mukaan tietty tunnus, minkä perusteella pelaaja voi erottaa toiset saman tyyppin ajoneuvot toisistaan. Pelaajan on myös mahdollista valita ajoneuvoihin oma maaliväri. Tunnuksen ja värin lisäksi asemalla määritellään ajoneuvoihin työkalut sekä henkilöstö. Eri ajoneuvoihin on mahdollista asettaa eri määriä työkaluja sekä henkilöstöä. Paljon käytössä olleita ajoneuvoja pelaajan täytyy myös huoltaa, muuten ajoneuvojen toiminta kärsii vaikuttaen pelaajan kykyyn suorittaa tehtäviä.

Henkilöstön palkkaaminen ja hallinnointi on myös tärkeää, pelaajan on mahdollista muuttaa henkilöiden kutsumanimet vapaasti halutessaan sekä valita henkilöille eri kykyjä tehtävillä saadun kokemuksen myötä, minkä johdosta henkilöstö toimii tehokkaammin tehtävillä, kykenevät käyttämään eri työkaluja tai suorittamaan uusia toimintoja, jotka eivät olleet aikaisemmin mahdollisia.



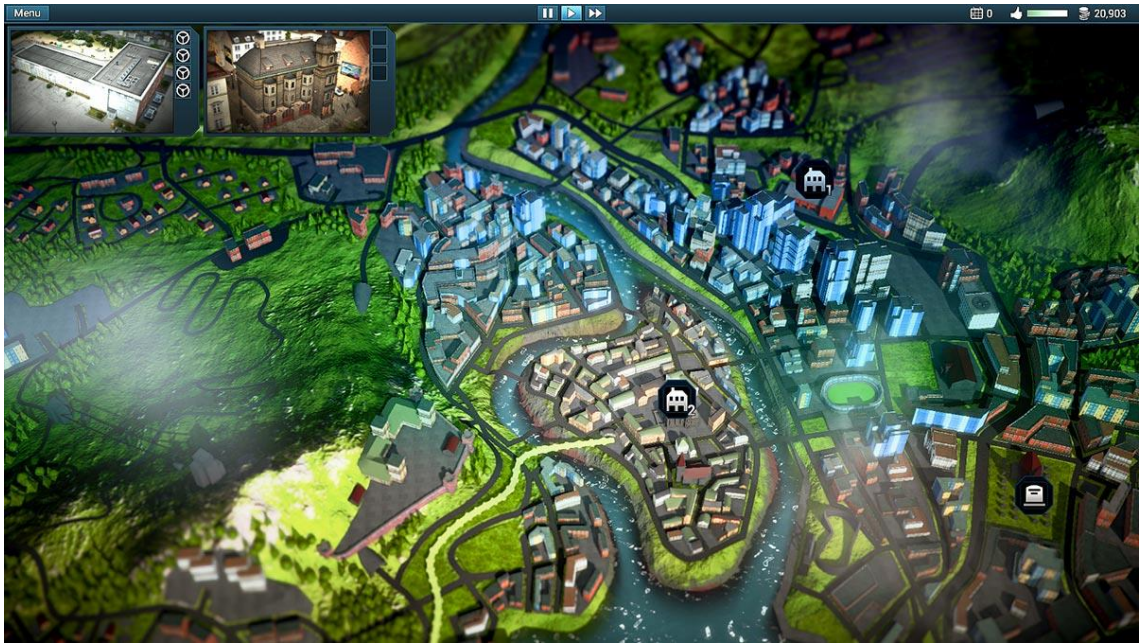
KUVA 1. Kuvakaappaus pelin asemapelitlasta, moderni paloasema.

3.1.2 Karttapelitila

Karttapelitila yhdistää asemapelitilan ja tehtäväpelitilan yhteen toimimalla linkkinä näiden kahden eri tilan välillä. Karttapelitila sisältää visuaalisen esityksen pelin kuvitteellisesta kaupungista, luoden identiteettiä kaupungille. Yksiköiden lähettäminen tehtäville tai asemille tapahtuu karttapelitilan kautta, missä pelaaja valitsee yksiköt sekä kohteen minne yksiköiden kuuluisi mennä.

Tehtävien ja asemien lisäksi kartalle ilmestyy erilaisia tapahtumia, jotka kertovat pelin kulusta ja pelaajan toiminnoista. Tapahtumat antavat pelaajalle palautetta toiminnasta sekä kuvaa pelin kaupunkia mutta ei vaikuta pelin toimintaan. Tapahtumiin kuuluu myös ilmoitukset uusista ajoneuvoista sekä työkaluista, mitkä ovat sen jälkeen saatavilla pelaajalle.

Uusien tehtävien ilmestyessä pelaaja saa ilmoituksen tehtävistä tekstinä ja selostuksena. Lähetyksen suoritettua pelaaja saa ilmoituksen lähetyksestä, minkä jälkeen ajoneuvot ilmestyvät pelin karttapelitilaan, jossa niiden kulkua voi seurata.



KUVA 2: Kuvakaappaus pelin karttapelitalasta, kuvitteellinen kaupunki Belvitsia.

3.1.3 Tehtäväpelitila

Pelituloista monimutkaisin on tehtäväpelitila, joka sisältää useita eri toiminnollisuuksia, joiden avulla pelin tehtäviä sekä pelaajan toimintoja hallinnoidaan. Pääsääntöisesti tehtävät jakautuvat kahteen eri kategoriaan; sammutustehtävät ja hätätapaukset, mutta haastavammat tehtävät sisältävät molemmat kategorian elementtejä, jolloin pelaajan on hallittava useita erilaisia pelastusyksiköitä. Suorittamalla tehtäviä pelaaja avaa lisää toimintoja, saa käytettäväksi enemmän resursseja sekä avaa haastavampia tehtäviä. Tehtävät pelissä jakautuvat satunnaisiintehtäviin ja tarinatehtäviin.

Hätätapaustehtävissä pelaaja hallitsee ensihoitoyksiköjä antaen ensiapua tehtävissä oleville uhreille ja kuljettaen ne sairaalaan tarvittaessa. Ensihoitoyksikköihin kuuluu ambulanssi, mihin pelaaja voi sijoittaa ensihoitajia sekä hoitotyökaluja, millä pelaaja voi antaa ensiapua uhreille tai parantaa pelaajan muuta henkilöstöä. Hoitamalla tai pelastamalla tehtävään liittyvät uhrin voi pelaaja suorittaa tehtävät onnistuneesti.

Sammutustehtävissä pelaajalla on käytettävissä laaja valikoima erilaisia ajoneuvoja sekä työkaluja, joita pelaajan täytyy käyttää oikein suorittaakseen tehtävän. Sammutustehtävissä pelaajan tavoitteena on sammuttaa tehtävällä olevat tulipalot. Tehtäväpelitalan tulipalot ovat vuorovaikutuksessa ympäristön kanssa, sillä tulipalot voivat levitä pelikentällä

eri paikkoihin sekä aiheuttaa sisätila tulipaloissa hapen puutetta vahingoittaen tilassa olevia yksiköjä. Tavallisten tulipalojen lisäksi, pelissä on erityyppisiä tulipaloja kuten rasvapalo sekä sähköpalo. Erityyppisten tulipalojen sammuttamiseen vaaditaan pelaajalta eri sammutusaineita, joita eri työkalut tuottavat. Pelaajan on myös mahdollista vaikuttaa tulipaloihin pelikentän kautta, esimerkiksi sammuttamalla sähkövirran palavista esineistä mahdollistaen erityyppisen sammutusaineen käytön palavaan esineeseen. Sammutus toimintojen lisäksi pelaajan täytyy navigoida pelikentällä eri esteiden läpi vaikuttamalla pelikentän eri esineisiin eri toiminnoilla.

Pelissä on myös tehtäviä, jotka sisältävät uhreja sekä tulipaloja, jolloin pelaajan on käytettävä ensiapu-, pelastus- ja sammutusyksiköitä tehtävien suorittamiseen. Palomiehet pelissä kykenevät liikkumaan haastavammissa olosuhteissa toisin kuin ensihoitajat, milloin pelaajan on käytettävä palomiehiä uhrien evakuointiin vaarallisista paikoista.

Suorittamalla tehtäviä pelaaja ansaitsee kokemuspisteitä sekä resursseja, riippuen kuinka monta yksikköä tehtävälle osallistui ja kuinka nopeasti tai hyvin pelaaja suoriutui tehtävästä. Kokemuspisteillä pelaajan yksiköt tulevat tehokkaammiksi ja tehtäviltä saatuja resursseja on mahdollista käyttää uusien ajoneuvojen ja työkalujen hankkimiseen.



KUVA 3: Kuvakaappaus pelin tehtäväpelitilasta, toimistotulipalo tehtävä.

3.1.4 Tutoriaalipelitila

Tutoriaalipelitila vastaa hyvin paljon pelin tehtäväpelitilaa, sillä erolla että pelaajalla on tehtävän sijasta tavoitteena opetella pelin ominaisuuksia suorittamalla tavoitteita tehtävällä. Pelaajalle annetaan tutoriaalitehtävän alkaessa tarvittavat yksiköt ja työkalut sekä rajoitetaan pelin komennot vain tarvittaviin komentoihin. Tehtävien tavoitteet sisältävät ohjeistuksia tekstin avulla ja autetaan pelaajaa korostamalla tehtävään liittyviä esineitä.

Tutoriaalitehtävät avautuvat pelin edetessä, joita pelaajan ei ole pakko suorittaa vaan voi ohittaa tehtävät ja edetä pelissä. Tehtävät ovat uudelleen pelattavissa pelin valikon kautta, milloin pelaaja voi opetella uudestaan pelin ominaisuuksia.

3.2 Projektin läpivienti

Projektin läpivientiä varten määriteltiin pelin projektisuunnitelmaan eri välietapit, joihin yritys sekä julkaisija sitoutuivat. Välietappien avulla pelin kehitystä oli mahdollista seurata verraten aiemmin määriteltyihin ominaisuuksiin ja sisältöön projektin edetessä. Myöhemmissä välietapeissa pelin ominaisuuksia sekä sisältöä voitiin arvioida pelin julkaisua varten, mahdollistaen tarvittavat muutokset pelin pelattavuuteen.

Peliprojektit jakautuvat tyypillisesti seuraaviin tuotantovaiheisiin: konseptointi, alkutuotanto, tuotanto ja jälkituotanto riippumatta minkälaisia menetelmiä projektinhallintaan käytetään (Keith, C. 2010). Projektissa ei ollut konseptointi vaihetta erikseen, sillä projektilla oli jo julkaisija ennestään sen ollessa suora jatko-osa aikaisemmalle projektille.

TAULUKKO 1: Projektin läpivientiä määritetyt välietapit (Sisäinen dokumentaatio).

Välietappi	Kuvaus
Ensimmäinen pelattava versio	Välietappi esittelee pelin pää ominaisuuksia eikä sisällä pelisisältöä taikka varsinaisia resursseja
Alpha	Välietappi, joka sisältää pelin tärkeimmät ominaisuudet, mutta pelin sisältö on hyvin rajallinen. Voi sisältää virheitä. Välietappiin yleisemmin kuuluu ominaisuuslukko milloin uusia ominaisuuksia ei tulisi määritellä peliin.
Beta	Versio pelistä, minkä pelaaja voi pelata alusta loppuun ja Alpha-version virheet on korjattu eikä tulisi sisältää mahdollisimman vähän virheitä.
Gold Master	Julkaisukelpoinen versio, joka ei saisi sisältää virheitä. Julkaisua varten on huomioita eri jakelukanavien tarpeet.

3.2.1 Alkutuotanto

Projektin tuotanto alkoi pelin alkutuotannosta, missä luotiin prototyyppejä pelin eri konaisuuksista selvittäen niiden vaativuutta, toimivuutta sekä hauskuutta. Alkutuotantoon kuuluu myös yleisemmin pelin eri ominaisuuksien ja sisällön määrittely tuotantoa varten, minkä avulla voidaan suunnitella projektin kulkua.

3.2.2 Ensimmäinen pelattava versio

Ensimmäinen pelattava versio välietappi sisälsi pelin ydin ominaisuudet kuten eri pääpe-
litilojen välillä navigointi, yksiköiden hallinta pelin asemapelitilassa, tehtävien käynnis-
täminen pelin karttapelitiassa sekä tehtäväpelitilan pääominaisuudet kuten tulisimulaa-
tion ja komennusjärjestelmän ensimmäiset versiot. Tehtäväpelitilassa oli toteutettuna
useita eri komentoja, mitä pelaaja pystyi käyttämään kuten sammutus, tikkaiden asennus
ja käyttö, letkuserivytysten tekeminen sekä kentässä hahmojen liikkuminen.

3.2.3 Alpha versio

Pelin alpha-versio keskittyi toteuttamaan pelin tärkeimmät ominaisuudet sekä sisältämään pelin sisältöä näiden ominaisuuksien toiminnan varmistamiseen. Alpha-version välietappi toi peliin laajennetun toiminnollisuuden pelin eri tiloihin, kuten asemapelitilan täysi toiminnollisuus, mikä sisälsi hahmojen palkkaamisen ja ajoneuvojen ostamisen lisäksi pelin hahmojen kykyjen kehittämisen, ajoneuvojen kaluston muokkaamisen, ajoneuvon värin vaihtamisen, hahmojen kutsumanimen vaihtamisen. Karttapelitilaan luotiin puuttuvat toiminnollisuudet, jotka sisälsivät yksiköiden lähettämisen eri kohteisiin, asemien avaamisen pelin edetessä sekä tehtävien synnyttämisen pelin etenemisen mukaan. Tehtäväpelitilaan toteutettiin puuttuvat yksikkökomennot, pelihahmojen vahingoittuminen, ensiapu yksiköt, uusia ajoneuvoja sekä päivitettiin pelin käyttöliittymää, jotta uudet ominaisuudet helposti käytettävissä.

Tarkoituksena oli toteuttaa alpha versioon kaikki ominaisuudet, minkä jälkeen pelin kehitys voisi keskittyä sisällön luomiseen, pelin tasapainottamiseen sekä virheiden korjaamiseen. Välietapin jälkeen kuitenkin todettiin, etteivät pelin ominaisuudet olleet ihan riittäviä, minkä johdosta välietapin jälkeen peliin toteutettiin vielä uusia ominaisuuksia.

3.2.4 Beta-versio

Beta-versioon määriteltiin toteuttavaksi pelin sisältö valmiiksi sekä alpha-versioon liittyvien virheiden korjaus sekä pelin tasapainotuksen muokkaaminen testauksen tuoman palautteen mukaisesti. Pääasiallisesti beta-version sisältö koostui pelin kenttien luomisesta sekä karttapelitilan kartan luomisesta. Kenttien lisäksi beta-versioon luotiin loput pelin tehtävistä, sisältäen satunnaiset-, tutoriaali- ja tarinatehtävät. Pelillisten ominaisuuksien lisäksi beta-versioon toteutettiin muiden ominaisuuksien sisältö kuten pelin äänimaailma, käännettävät tekstit, pelitilan tallentaminen ja lataaminen. Versioon myös lisättiin tuki Steam-palvelun eri toiminnoille kuten pelin saavutukset.

Tasapainotuksessa pyrittiin varmistamaan pelin hauskuus eri tavoilla. Erityistä huomiota kiinnitettiin pelin pelattavuuteen sekä pelin vaikeustason nousuun pelin edetessä. Vaikeustasoa nostamalla pyrittiin kannustamaan pelaajaa hyödyntämään pelin eri ominaisuuksia, tehden pelikokemuksesta mielenkiintoisempaa.

Aikaisempien ongelmien takia pelin sisältö ei ollut valmis alkuperäistä beta-välietappia varten, jonka takia välietappia siirrettiin myöhempään ajankohtaan, jotta pelin sisältö saataisiin riittävälle tasolle. Lisäajan johdosta pystyttiin peliä tasapainoittamaan enemmän sekä korjaamaan lisää virheitä.

3.2.5 Gold Master versio

Tuotannon viimeinen välietappi Gold Master sisälsi version pelistä mikä voitiin viedä jakeluun. Versio kuului pelin kaikki sisältö ja ominaisuudet ilman virheitä, jotka haittasivat huomattavasti pelaamista.

Julkaisuun liittyvät merkittävät ominaisuudet sekä sisältö:

- 8 Ajoneuvoa
 - kevyt paloauto, paloauto, ambulanssi, vaahtotykki auto, jauhettykkiauto, vesitykkiauto, tikasauto ja vesitankkiauto
- 2 Hahmotyyppiä
 - palomies ja ensihoitaja
- 20 Eri taitoa yksiköille
- 3 Paloasemaa
- 9 Tehtäväkenttää
- 24 Satunnaistehtävää
- 12 Tarinatehtävää
- 13 Tutoriaalitehtävää
- 9 Käännettyä kieltä
 - englanti, saksa, ranska, italia, espanja, turkki, puola, venäjä, Brazilian portugali
- 20 Saavutusta
- 3 Erityyppistä tulipaloa
- Tulisimulaatio sisä- ja ulko tuli paloille

3.2.6 Julkaisu

Peli julkaistiin 3.6.2015 Windows- ja Mac OS X-alustoille Steam-palvelussa sekä erinäississä verkkokaupoissa. Julkaisua ennen peli oli mahdollista ostaa alennettuun hintaan ennakotilauksena Steam-palvelusta. Kivijalkakaupoista tai verkkokaupoista oli myös mahdollista ostaa pelistä keräilyversio (engl. Collector's Edition), joka sisälsi pelin lisäksi oheistuotteita kuten pelin logon pohjalta luotu metallinen mitali, palomies paperinukke, juliste, kuvituksia pelistä (3kpl) sekä erilaisia peli teemaisia tarroja.

TAULUKKO 2: Julkaisua varten määritetyt järjestelmävaatimukset alusta kohtaisesti.

	System Requirements			
	Windows		MAC OS X	
	Minimum	Recommended	Minumum	Recommen- ded
OS	Windows Vista/7/8/10	Windows 7/8/10 (64-Bit)	MAC OS X 10.8	MAC OS X 10.8
Proces- sor	Quadcore 2,4 GHz; AMD Phenom II X3 720 / Intel Core2 Quad Q6600	AMD FX-8350 4,0 GHz / Intel Core i7-3820 3,60GHz	Intel Core i5 or i7 processor	Intel Core i5 or i7 proces- sor
Memory	4 GB RAM	8 GB RAM	4 GB RAM	8 GB RAM
Graphics	AMD Radeon R7 250 / GeForce GTX 650	Nvidia GTX 760 / Radeon R9 255	ATI Radeon HD 5750 with 1 GB VRAM or higher	NVIDIA GeForce GTX 750M
DirectX	Version 9.0c	Version 9.0c		
Hard Drive	1 GB available space	1 GB available space	1 GB available space	1 GB available space
Sound Card	sound card	sound card	sound card	sound card
Additi- onal No- tes	Input devices: Keyboard and mouse	Input devices: Keyboard and mouse	Mouse and keyboard	Mouse and keyboard

3.2.7 Jälkituotanto

Pelin julkaisun jälkeen sai yritys toimekseen luoda korjausversion pelaajien palautteen perusteella. Palautteesta kerättiin tärkeimmät asiat, mistä julkaisijan ja kehitystiimin yhteistyönä valittiin korjattavat seikat tärkeyden sekä toteuttamiskelpoisuuden perusteella.

Korjattaviin ja muutettaviin asioihin kuului: pelin tasapainotuksen muokkaaminen, jotta pelin eteneminen kuluisi nopeammin ja sulavammin, pelin tehtävien satunnaistamisen muokkaaminen siten että aiemmin pelatut tehtävät ilmestyisivät huomattavasti harvemmin, ominaisuuksien parantaminen kuten komentojen suorittaminen polunetsinnän yhteydessä, pelin kulkua estävän tikasautoon liittyvän virheen sekä useita pieniä parannuksia ja korjauksia.

Korjausversion jälkeen julkaisija tilasi myös lisää kielikäännöksiä peliin, jolloin myös luotiin tuki Apple-yrityksen App Store palvelu jakelua varten. Uudet kielikäännökset sisälsivät tsekin ja unkarin kielen. Kielikäännökset päivitettiin myös Steam-palvelussa jaettuun versioon.

4 PROJEKTIN TOTEUTUS

4.1 Suunnittelu

Pelin tuotannon alkaessa projektin läpivientiä varten luotiin pelisuunnitteludokumentti (GDD), joka kuvailee pelin pääominaisuuksia yksityiskohtaisesti antaen edellytykset projektiryhmälle luoda pelin ominaisuudet ja sisältö suunnittelun mukaisesti. Pelisuunnitteludokumentti toimii ohjeena henkilöstölle tuotannossa, johon tiimin henkilöt voivat tukeutua tuotannon aikana.

Suunnittelun ohessa pelin keskeisiä toiminnollisuuksia kokeiltiin prototyyppien muodossa yrittäen selvittää mahdollisia toteutustapoja, esteitä, mahdollisuuksia tai rajoitteita. Prototyypit yleisemmin sisälsivät jonkin ennalta määritetyn pelin ominaisuuden yksinkertaisen toteutuksen, minkä avulla pelin hauskuutta voitiin testata ennen pelin tuotantoa sekä toimien apuvälineenä pelin suunnittelussa.

Projektin läpivientiä varten määriteltiin välietappisuunnitelma. Välietappisuunnitelma määrittää pelin tuotannon kannalta eri asioiden aikataulut suhteessa toisiinsa, ottaen huomioon mahdollisia riippuvuuksia eri osa-alueiden kanssa. Välietappisuunnitelma toimi myös osana julkaisijan kanssa tehtyjä sopimuksia.

Projektin aikataulutuksessa on vaikeaa arvioida eri tehtävien kestoa tarkasti, sillä kestoon vaikuttavat useat eri tekijät, joista osa voivat olla tuntemattomia hyvinkin pitkään. Käytetyt työkalut, henkilöiden kokemus, resurssit sekä muutokset pelisuunnitteluun tekevät tehtävien arvioinnista hyvin vaikeaa. Projektin alussa pelin suunniteltu skaala oli liian suuri saatavilla olevien resurssien suhteen, minkä johdosta pelinsuunnittelua piti muuttaa tuotannon aikana huomattavasti aiheuttaen ongelmia aikataulutuksessa. Ylioptimistiset aikataulut ovat yleinen ongelma miksi pelien tuotannot ajautuvat ongelmiin (Keith, C. 2010).

4.2 Projektin kulku

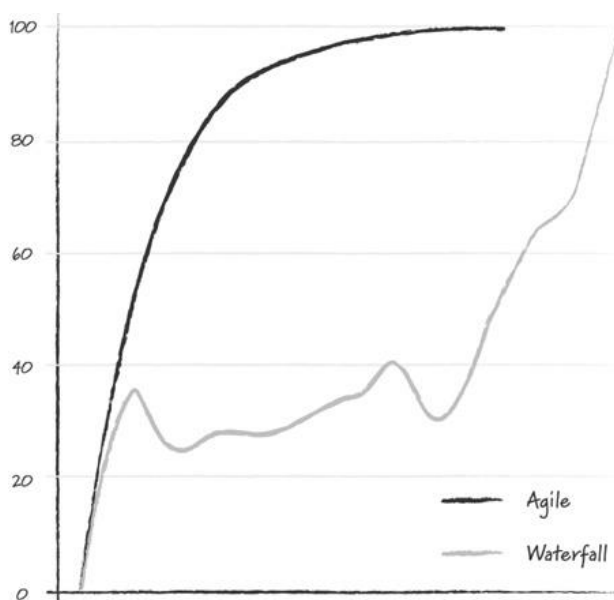
Projektin tuotanto jakautui useisiin välietappeihin, missä suurimpina muutoksina olivat kehitystiimin roolien ja henkilöstömäärän muuttuminen sekä toteutettavat asiat. Projektin kulun seuraamista varten projektin edetessä pidettiin useita palavereja kuukaudessa yrityksen sekä julkaisijan kanssa, missä selvitettiin tuotannon aikana ilmeneviä ongelmia.

4.2.1 Projektinhallinta

Projektin henkilöstöön kuului hyvin monen eri osa-alueen tekijää, minkä johdosta eri tehtävät jouduttiin antamaan hyvinkin tarkoin määritetyille tekijöille, sen sijaan että tiimi kaikki jäsenet olisivat voineet suorittaa mitä tahansa tehtäviä. Suurimmat osa-alueet jakautuivat ohjelmointiin, grafiikan luontiin, pelisuunnitteluun, kenttäsuunnitteluun ja tuottamiseen. Tyypillisemmin osa-alueet pitivät sisällään yhdestä kolmeen työntekijää riippuen tuotannon vaiheesta.

Projektissa järjestettiin päivittäisiä aamupalavereja Scrum-tyyppisesti, missä työntekijät kertoivat aikaisemmat ja tulevat tehtävät sekä mahdolliset esteet tuleville tehtäville. Aamupalaverien johdosta saatettiin järjestää erillisiä palavereja, jotka käsittelivät esille nousseita asioita. Julkaisija kommunikation ylläpitämiseksi järjestettiin lähes viikoittain julkaisijan kanssa telekonfrensseja, joissa käytiin läpi projektin kulkua sekä arvioitiin viimeisimpää välietappia.

Peleille ominaista on, ettei pelin pelattavuudesta tai hauskuudesta voida olla täysin varmoja, ellei olla tekemässä hyvin samantyylistä peliä kuin mitä on aikaisemmin tehty. Ketteriä menetelmiä soveltamalla voi peleistä löytää hauskuuden nopeammin. Iteroinnin avulla voidaan löytää pelin hyvät ja huonot ominaisuudet aikaisin tuotannossa.



KUVA 4: Hauskuuden löytäminen pelissä (Keith, C. 2010).

Tuotteen kehitysjono (engl. Product Backlog) on järjestetty lista pelin ominaisuuksista tai vaatimuksista (Keith, C. 2010). Projektissa tuotteen kehitysjono toteutettiin heikosti eikä kattavaa listaa esineistä pidetty tarpeeksi hyvin ajan tasalla. Listasta saattoi puuttua esineitä, mikä johti siihen, että projektin kokonaisuutta oli vaikea seurata. Tulevissa projekteissa tulee pitää huolta tuotteen kehitysjonosta pitämällä sen ajan tasalla ja helposti saatavilla, jolloin projektin kehitystä voidaan seurata ja hallita paremmin. Atlassianin yrityksen luoma JIRA Agile voi olla hyvä työkalu tulevia projekteja varten, koska yrityksen henkilöstöllä on jo kokemusta muista Atlassianin yrityksen työkaluista.

4.2.2 Versionhallinta

Projektin alkaessa oli selvää, että kehitystä varten tarvitaan versionhallinta kehityksen seuraamista ja turvaamista varten. Aikaisemmissa projekteissa oli käytössä Unityn tarjoama versionhallinta, mikä todettiin projektien aikana huonoksi valinnaksi käytettävyyden sekä luotettavuuden kannalta. Versionhallinta järjestelmästä Mercurial järjestelmä vaikutti hyvältä projektin tarpeisiin sekä Atlassianin-yrityksen tarjoama Bitbucket palvelu tarjosi edullisen vaihtoehdon versionhallinnan tiedostojen säilyttämiseen.

Mercurial on hajautettu versionhallinta järjestelmä, missä käyttäjien on mahdollista luoda omia versionhallinta haaroja, jotka voi myöhemmin yhdistää päähaaraan tai muihin haaroihin, minkä avulla voi kehittäjä voi toteuttaa tarvittavaa toiminnollisuutta vaikuttamatta muiden kehittäjien toimintaan. Versionhallinnan avulla voitiin olla varmoja, että pelistä

löytyy toimiva versio aina tarvittaessa, jos sellainen on olemassa versionhallinnan historiassa.

Mercurial versionhallinta palveli projektin tarpeita erinomaisesti, vaikkakin on hieman vaikeampi käyttöinen kuin Unityn tarjoama versionhallintaratkaisu. Mercurial on pääsääntöisesti suunniteltu vain ohjelmakoodin versionhallintaa, mutta projektissa pystyttiin silti muutkin resurssit säilyttämään versionhallinnassa ilman ongelmia. Edullisuuden ja toimivuuden puolesta ei olisi syytä vaihtaa versionhallintaa toiseen tulevilla projekteilla.

4.2.3 Testaus ja virheiden seuranta

Pelin julkaisussa on tärkeää, ettei peli sisällä virheitä. Digitaalisen jakelun yhteydessä virheiden korjaaminen julkaisun jälkeen on kuitenkin mahdollista, mutta voi aiheuttaa suurta vahinkoa pelille, jos se sisältää virheitä, jotka vaikuttavat pelaajan pelikokemukseen. Virheiden löytämistä varten tarvitaan pelitestaajia, joiden tarkoituksena on pelata peliä, löytää ja raportoida virheitä joita kehittäjä voi saadun tiedon perusteella ratkaista.

Virheiden seurantaan projektissa käytettiin kahta eri virheiden seuranta sovellusta. Atlasian JIRA (joka oli myös käytössä osaksi projektin hallinnassa), mitä käytettiin yrityksen oman testauksen puolesta. Mantis Bug Tracker-sovellusta käytettiin yhteistyössä julkaisijan kanssa. Molempien sovellusten toimintaperiaate on hyvin samanlainen ja molemmat toteuttavat lähes samat ominaisuudet.

TAULUKKO 3: Virheiden kategoriat ja kuvaukset projektin toteutuksessa

Kategoria	Kuvaus
Estävä	Pelin kulkua tai pelaamista estävä virhe, minkä johdosta peliä ei voida pelata suunnitellusti.
Kriittinen	Virhe joka estää pelin ominaisuutta toimimasta odotetulla tavalla muuttaen pelikokemusta tai rikkoen suunniteltua toimivuutta.
Vähäinen	Ei vaikuta pelikokemukseen, mutta voi olla häiritsevä tekijä.
Muutos	Pelin ominaisuutta tai sisältöä koskeva muutospyyntö.

Projektissa laadunvarmistamista varten oli projektissa muutamia pelitestaajia, joiden tehtävänä oli pelata peliä, raportoida virheitä ja työskennellä yhdessä kehitystiimin kanssa

pelin hauskuuden varmistamiseksi. Virheitä kohdatessaan testaajat raportoivat virheet käytettyyn virnehallinta sovellukseen (JIRA), minkä jälkeen virheet määrättiin eri kehittäjille virheen luonteen mukaisesti. Myöhemmin tuotannossa testaajat saattoivat pystyä itse määrittämään virheet tietyille kehittäjille, kun osa-alueet tuotannossa olivat selkeämpiä. Tämän johdosta projektipäällikön tai muun johtavan työntekijän ei tarvinnut kuluttaa aikaa virheiden määrittämiseen. Virheiden raportoinnin ja määrittämisen jälkeen kyseinen kehittäjä korjasi virheen ja palautti virheen testaajalle tarkistettavaksi. Tarkistuksen jälkeen virheet suljettiin tai määrättiin takaisin kehittäjälle, riippuen oliko virhe ratkaistu.

Oman pelitestauksen ohella, julkaisija järjesti pelitestausta ulkoisilla testausyrityksillä, joista kehittäjät saivat löydetty virheet sekä kattavan raportin testausprosessista ja pelin tilasta. Ulkoisessa pelitestauksessa jäi virheiden määrittäminen täysin projektipäällikölle ja muille vastuullisille kehittäjille.

Pelien laadunvarmistus on haastava osuus tuotannossa, mikä vaatii ammattitaitoisia pelitestaajia sekä prosessin hallintaa avustavia järjestelmiä. Projektissa käytetty JIRA sovellus vastasi projektin tarpeisiin keskinkertaisesti. JIRA on laaja ominaisuuksiltaan ja sallii käyttäjien muokata sen toteutusta hyvin laajasti, minkä avulla sovelluksesta voidaan luoda hyvin tilanteeseen sopiva. Projektin virheiden hallintaa varten JIRA:n toiminnallisuutta kutistettiin tehden sovelluksen käytöstä yksinkertaisempaa, esimerkiksi poistamalla eri tehtäväkategorioita käytöstä. Tulevissa projekteissa JIRA voi olla erinomainen työkalu virheiden seurantaan, mutta sen toiminallisuutta on syytä muokata projektin tarpeisiin sopiviksi.

4.3 Keskeisten ominaisuuksien toteutus

Peli sisälsi paljon erilaisia ominaisuuksia, mistä osa oli muita huomattavasti tärkeämpiä. Pelikokemuksen kannalta jotkut ominaisuudet suunniteltiin ja toteutettiin hienovaraisesti, sillä niiden painoarvo oli suurempi muihin ominaisuuksiin nähden. Tehtäväpelitila oli pelin tiloista haastavin. Tilaan kuului useita eri ominaisuuksia sekä järjestelmiä. Näiden täytyi toimia hyvin yhdessä, jotta pelikokemus olisi miellyttävä.

4.3.1 Tulisimulaatio

Tehtävien kannalta tulisimulaatio oli tärkein ominaisuus pelissä, missä tulen palamista, leviämistä ja sammumista simuloidaan eri arvojen avulla. Eri objektit palavat eri tavalla riippuen niiden massasta, materiaalista, etäisyydestä toisiin objekteihin, hapensaannista sekä ympäristön aiheuttamista erikoistilanteista. Tarkoituksena oli luoda simulaatio, joka tarjoaa pelaajalle informaatiota tilanteesta sekä useita eri keinoja lähestyä tulipaloja sekä vaikuttaa tulien toimintaan.

Tulipalot käyttäytyivät eri tavalla sisä- ja ulkotiloissa, missä tulen hapen saanti muuttuu huomattavasti, milloin tulipalot voivat jäädä kytemään ja syttyä räjähdysmäisesti uudelleen, jos ympäristö muuttuu pelaajan toimesta. Tulipalojen tyypit vaihtelevat palavan objektin mukaan, milloin pelikentällä saattoi olla sähköpaloja ja rasvapaloja tavallisten tulipalojen lisäksi. Pelaajan on mahdollista lähestyä erityyppisiä tulipaloja eri tavalla, sillä eri sammutusaineet vaikuttavat erityyppisiin tulipaloihin eri tavalla.

Aluksi tulipalon simuloinnissa pyrittiin liian realistiseen lopputulokseen. Tämä aiheutti ongelmia ominaisuuden kommunikoinnissa pelaajalle, sillä pelaajan oli vaikea ymmärtää pelin tapahtumaketjuja. Liian tarkka simulointi aiheutti ongelmia myös tehtävien suunnittelussa milloin kenttäsuunnittelijan oli vaikeaa hallita tehtävän kulkua. Tuotannon lopussa todettiin että simulaatiota pitää hieman yksinkertaistaa, jotta tehtävistä saadaan mielenkiintoisia.

Keskittymällä enemmänkin tulen simuloimiseen pelaajan näkökulmasta, olisi toteutusta voinut muokata yksinkertaisemmaksi ja luoda enemmän vuorovaikutuksia ympäristön kanssa tehden sammutustehtävistä vielä mielenkiintoisempia. Liian yksityiskohtainen suunnittelu tulen simuloinnissa aiheutti tuotannossa pitkään ongelmia, mutta tuotannon lopussa saatiin ominaisuudesta hyvä kokonaisuus.

4.3.2 Komentojärjestelmä

Pelaaja suorittaa tehtäviä komentamalla eri yksiköitä suorittamaan eri toimintoja. Komennot koostuvat tehtävistä, jotka määrittävät komentojen toiminnollisuuden. Komentojen

tehtäviä on helppo muokata tarpeen mukaan lisäämällä tai poistamalla tehtäviä komennosta. Esimerkiksi sammutus komento koostuu liike-, tarkistus- sekä sammutustehtävistä.

Pelimaailmassa sijaitsee komentokohteita, joihin pelaaja voi komentaa yksiköitä. Komennettavien yksiköiden määrä rajattiin yhteen, sillä valtaosa komennettavista kohteista sallivat vain yhden yksikön. Komennettavien yksiköiden määrä rajattiin yhteen, sillä valtaosa komennettavista kohteista sallivat vain yhden yksikön. Yleisimpä komentoja varten olisi kuitenkin voitu tehdä poikkeus, mutta projektin tuotannon alussa päätettiin pitää komennot yhteneväisinä. Komentokohteilla on konteksti komennot, milloin pelaajan oli helppo komentaa yksiköt kohteisiin toimimaan odotetulla tavalla valmistamatta komentoa pelin komentopaneelist.

Komentojärjestelmän toteutus onnistui projektissa hyvin, sillä pienen opettelun jälkeen pelaajan oli helppo ymmärtää kuinka järjestelmä toimi sekä uusien komentojen luonti tapahtui helposti, milloin peliä oli helppo laajentaa ja muokata palautteen mukaan.

4.3.3 Yksiköiden hallinta

Pelin yksiköiden hallinta kattaa kokonaisuudessaan useita eri osa-alueita kuten yksiköiden palkkaaminen ja erikoistuminen, ajoneuvojen ostaminen sekä työkalujen ja henkilöstön hallitseminen. Yksiköiden hallinnan avulla pelaajan oli mahdollista luoda ryhmiä erilaisia ajoneuvoja sekä yksikkö yhdistelmiä, minkä johdosta pelaaja pystyi varautumaan hyvinkin erilaisiin tehtäviin.

Tehtäviä suorittamalla pelaaja palkittiin erilaisilla resursseilla, mitä pelaajan oli mahdollista käyttää haluamallansa tavalla. Pelin rahaa oli mahdollista käyttää uusien ajoneuvojen, paloasemien sekä työkalujen hankkimiseen ja henkilöstön palkkaamiseen. Kokemuspisteitä keräämällä pelaajan oli mahdollista luoda erikoistuneita yksiköitä, mitkä saattoivat suoriutua muita yksiköitä paremmin tietyistä tehtävistä.

Pelin tehtävät jäivät toiminnollisuudeltaan yksinkertaisiksi, minkä takia yksiköiden hallitsemisella ei ollut suurta merkitystä pelin kulussa. Yksiköiden hallinta kuitenkin mahdollisti hyvinkin erilaisia tapoja suorittamaan tehtäviä, tehden pelaamisesta monipuolista.

4.3.4 Pelin eteneminen

Pelin etenemistä varten luotiin järjestelmä, joka arpoo pelaajalle tehtäviä riippuen pelin tilanteesta. Etenemistä valvova järjestelmä myös palkitsee pelaajan avaamalla uusia ominaisuuksia sekä antamalla resursseja pelin edetessä kehittäjien määrittelemällä tavalla.

Järjestelmän avulla voitiin pelin etenemistä muokata saadun palautteen mukaan hyvinkin nopeasti. Toteutus onnistui hyvin, sillä etenemisen muokkaaminen oli hyvin tärkeää pelin pelattavuuden kannalta ja järjestelmä ei asettanut mitään rajoitteita etenemisen muokkamiseen. Pelin ensimmäisessä julkaistussa versiossa järjestelmä saattoi arpoa saman tehtävän vain yhden eri tehtävän ollessa välissä. Vika kuitenkin korjattiin pelin jälkituotannossa.

4.4 Unity-laajennukset

Unityn Asset Store kauppapaikka tarjoaa kehittäjille mahdollisuuden ostaa muiden Unity kehittäjien tekemiä laajennuksia, minkä johdosta pelin kaikkia ominaisuuksia ei tarvitse täysin toteuttaa itse. Laajennuksia on mahdollista löytää myös muualtakin kuin Asset Store palvelusta, kuten yksittäisten Unity kehittäjien kotisivuilta.

Laajennusten käyttäminen vapautti paljon projektin resursseja, milloin oli mahdollista keskittyä pelin keskeisiin ominaisuuksiin. Laajennusten käyttöönotossa on kuitenkin huomioitava, milloin kyseinen laajennus on tehty ja onko sitä päivitetty käytettyyn Unity versioon. Tuotannossa sattui tilanteita, joissa käytetty laajennus ei toiminut kyseisen Unity version kanssa tai laajennukseen oli tehtävä korjauksia, jotta sitä olisi mahdollista käyttää.

Osa projektin käytetyistä liitännäisistä otettiin käyttöön, sillä ne oli todettu toimiviksi käyttötarkoituksiin aikaisemmissa projekteissa eikä lisäominaisuuksia tarvittu. Vaihtoehtoisia liitännäisiä löytyy toteutettuihin ominaisuuksiin, mutta tuotannon aikana todettiin valitut liitännäiset riittäviksi projektin tarpeisiin.

4.4.1 Localization Package

Pelien kääntäminen eri kielille on yleinen käytäntö, millä pyritään laajentamaan pelin saatavuutta eri maissa. Lokalisoinnilla yleisemmin käännetään pelin tekstit eri kielille, mutta se voi myös sisältää eri kuvien tai äänien kääntämistä. Unity ei tue oletuksena lokalisointi ominaisuutta, jolloin kehittäjän on itse luotava tuki lokalisoinnille.

M2H Game Studio-yrityksen kehittämä Localization Package-liitännäinen toteuttaa lokalisoinnin Unity ympäristössä yksinkertaisen rajapinnan kautta, minkä avulla kehittäjän on helppo luoda tuki useille eri kielille. Liitännäinen käyttää toiminnassaan Googlen Google Drive-palvelua, minne kehittäjä luo taulukkotiedoston, joka sisältää käännettävän kieli-tiedon ja mistä liitännäinen hakee tiedon automaattisesti Unity ympäristöön.

Projektin tekstit luotiin alkuperäisesti englanniksi, käännettiin projektin aikana eri kielille kuten saksa, ranska, italia, espanja, puola, brasilianportugali, venäjä ja turkki. Liitännäisen yksinkertaisuuden vuoksi oli helppoa luoda erilaisia Unity komponentteja, joita kuka tahansa projektin henkilöstöstä pystyi käyttämään tarvitessaan käännettävää tekstiä esimerkiksi käyttöliittymään

		EN	DE	FR
	Comments	578		
GAME_TITLE	0	Rescue 2	Rescue 2	Rescue 2
NEW_GAME		New Game	Neues Spiel	Nouvelle partie
RESUME_GAME		Resume Game	Spiel fortsetzen	Reprendre la partie
LOAD_GAME		Load Game	Spiel laden	Charger une partie
SAVE_GAME		Save Game	Spiel speichern	Sauvegarder la partie
OPTIONS		Options	Optionen	Options
QUIT		Quit	Beenden	Quitter

KUVA 5: Kuvakaappaus pelin käänntiedosta, jossa eri sarakkeisiin on määritetty eri kielten käännökset vastaavaa avainarvoa varten.

Kaikkiaan liitännäinen toimi hyvin projektin tarpeita varten. Pieniä ongelmia aiheuttivat tekstikentät, joiden ensimmäinen merkki oli jokin matemaattinen merkki, minkä takia tekstien sisältö muuttui automaattisesti taulukko funktioksi aiheuttaen virheellistä tietoa. Vastaavia toteutuksia on helppo löytää, mutta liitännäinen oli todettu aikaisemmissa projekteissa tarpeeksi hyväksi, joten syytä ei ollut vaihtaa muuhun toteutukseen. Tuleviin projekteihin tulee arvioida olisiko lokalisointi liitännäisen syytä toteuttaa myös muita ominaisuuksia, milloin voi olla syytä vaihtaa liitännäistä.

4.4.2 NGUI (Next-Gen UI)

Graafinen käyttöliittymä on keskeinen osa tietokonepelejä. Se koostuu teksteistä, kuvista sekä käyttöliittymä elementeistä, joiden perusteella käytetään tietokonepeliä. Käyttöliittymäelementteihin kuuluu muun muassa ikkunoita, valikoita, valintanappeja ja valintalaatikoita. Riippuen peligenrestä ja tietokonepelin tyylistä on graafisen käyttöliittymän painoarvo vaihteleva.

Tasharen Entertainment-yrityksen kehittämä Next-Gen UI kit-liitännäinen sisältää lukuisia ominaisuuksia, joiden avulla kehittäjä pystyy luomaan käyttöliittymän elementtejä Unity ympäristöön WYSIWYG (lyhenne sanoista What You See Is What You Get) periaatteen mukaisesti. Elementtien luonnin lisäksi, liitännäinen suoriutuu graafisen käyttöliittymän piirrossa huomattavasti Unityn oletus toteutusta paremmin sekä mahdollistaa kehittäjän tekemään erilaisia lisätoimintoja suorituskyvyn parantamiseksi. Liitännäiseen on myös toteutettu tuki Unity-käyttöliittymän kautta ominaisuuksien hallintaan, milloin kehittäjän on helppo toteuttaa haluttuja toimintoja intuitiivisesti.



KUVA 6: Atlas-kuva pelissä käytetyistä komento ikoneista käyttöliittymän piirtoa varten.

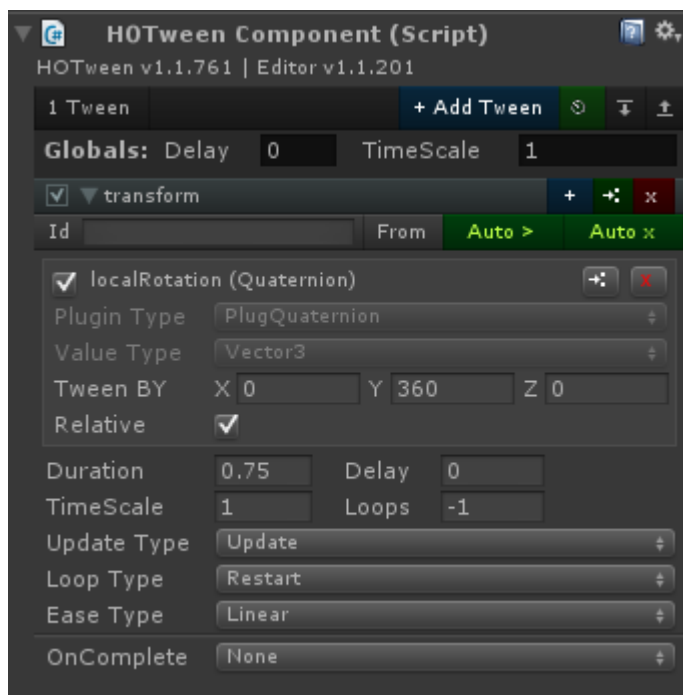
Liitännäinen toteutti tarvittavat ominaisuudet erinomaisesti projektin tarpeita varten, minkä johdosta käyttöliittymä oli helppo luoda ja muokata myöhemmin. Satunnaiset atlasen luonti virheet aiheuttivat pieniä ongelmia projektissa, mutta ongelman tiedostettua olivat virheet vaivatonta korjata. Liitännäisen tehokas käyttö vaatii kokemusta, jotta kaikkia toteutettuja ominaisuuksia voi hyödyntää tehokkaasti. Vartenotettava vaihtoehto liitännäiselle on Unity 4.6 versiossa julkaistu oma toteutus käyttöliittymästä, jota tulisi testata tuleviin projekteihin ja arvioida kuinka hyvin tarvittavat toiminnollisuudet toteutuvat. Jos Unityn oma toteutus jää heikoksi käyttöliittymän osalta, on syytä käyttää NGUI-liitännäistä jatkossakin.

4.4.3 HOTween

Tween-animaatiot käsittävät erityyppisten arvojen muuttamista valittujen arvojen välillä ajan kuluessa. Tween-animaatiot soveltuvat varsinkin yksinkertaisten muutosten tekemiseen. Monipuolisempien animaatioiden luomiseen voidaan yhdistää useita animaatioita, jotka sovitetaan yhtä aikaisesti.

Daniele Giardinin kehittämä HOTween-liitännäinen tarjoaa rajapinnan, millä Tween-animaatioita voi laukaista yksinkertaisilla ohjelmakomennoilla, käyttäen liitännäisen hallinnoimaa Tween-moottoria hyvän suorituskyvyn saavuttamiseen. Ohjelmakomentojen lisäksi liitännäinen tarjoaa visuaalisen käyttöliittymän Tween-animaatioiden laukaisemiseen, missä mahdolliset toiminnollisuudet ovat rajoitetumpia.

Projektissa liitännäistä käytettiin erityisesti graafisen käyttöliittymän elementtien hallitsemiseen, sillä muutokset niissä olivat yksinkertaisia toteuttaa, mutta käyttöä löytyi myös pelimaailman objekteille, kuten karttatilassa ajoneuvojen liikuttamiseen tai tehtävätilan ajoneuvojen hyttien tai lokerojen ovien aukaisemiseen ja sulkemiseen.



KUVA 7: Tween-komponentin graafinen käyttöliittymä. (Ilmastolaitteiden pyörittämis-animaatio).

Liitännäiselle löytyy paljon eri vaihtoehtoja, luvussa 4.3.2 mainittu NGUI-liitännäinen toteuttaa perus tween-animaatioita sen käyttämiä käyttöliittymä komponentteja varten. Aikaisemmissa projekteissa käytetty iTween-liitännäinen olisi toiminut vastaavasti kuin käytetty liitännäinen, mutta HOTween on parempi suorituskyvyltään kuin iTween, vaikkakin hieman vaikeampi ottaa käyttöön. Mikään ei myöskään estä kehittäjää käyttämästä useampia liitännäisiä toteuttamaan samaa toiminnollisuutta, joten on mahdollista käyttää useita tween-animaatioiden toteutuksia projekteissa. Mobiililaitteille kehittäessä on syytä huomioida toteutusten suorituskyky. Muissa tilanteissa on hyvin yhdentekevää, mikä toteutus valintaan projektia varten.

4.4.4 Ludosity Steamworks Wrapper

Pelien digitaalinen jakelu on kasvanut viime vuosien aikana. Valve Corporation-yrityksen Steam-palvelu on yksi merkittävimmistä digitaalisista jakelukanavista. Steamworks-palvelu on luotu kehittäjille, jotka haluavat julkaista tietokonepelejä Steam-palvelussa. Julkaisemisen lisäksi Steam mahdollistaa useita eri toimintoja, mitä kehittäjä voi toteuttaa tuotteisiinsa. Näistä yleisimmät ovat erilaisten tilastojen tarkastelu, saavutusten ansaitseminen ja keräilykorttien jakaminen. Toimintojen toteuttamiseen vaaditaan kehittäjältä Steamworks rajapinnan käyttämistä ominaisuuksien toteuttamiseksi.

Ludosity's Steamworks Wrapper on Ludosity-yrityksen kehittämä liitännäinen. Sen avulla voidaan toteuttaa Unity ympäristössä Steamworks-palvelun ominaisuuksia, käyttämällä Unityn omaa skriptauskieltä. Liitännäisen avulla toteutettiin muutamia Steam-palvelun ominaisuuksia kuten pelin saavutusten hallinta (engl. Achievements).

Projektin Steam-ominaisuuksien toteuttaminen oli vaivatonta käytetyllä liitännäisellä. Huomioitavaa on että myöhemmissä Steamworks SDK julkaisussa on lisätty C#-kielen sidokset, milloin liitännäiselle ei välttämättä ole tarvetta. Tuleviin projekteihin on syytä varmistaa, että Steamworks SDK toteutus toteuttaa tarvittavat ominaisuudet Steam-palvelua varten.

4.4.5 MasterAudio

Äänimaailma on yksi tärkeä elementti peleissä riippuen tietenkin pelisuunnittelusta. Yleisemmin peleihin liittyy musiikkia, ambienssia sekä erityyppisiä ääniefektejä. Äänet luovat tunnelmaa sekä voivat toimia vahvistavina keinoina pelin kerronnassa sekä ohjauksessa antaen tosiaikaista palautetta pelaajan toiminnoista.

Dark Tonic Games-yrityksen kehittämä MasterAudio-liitännäinen toteuttaa useita eri ominaisuuksia äänien käsittelyssä, tehden äänimaailman luomista peliin kehittäjälle helpompaa. Ominaisuuksiin kuuluu äänien satunnaistaminen ja ryhmittäminen, soittolistojen luominen, äänien vaimentaminen sekä äänien ristihäivyttäminen, äänen tason muuttamiseen ja kehittäjälle luotu graafinen käyttöliittymä äänien hallitsemiseen. Liitännäinen tarjoaa myös ohjelmarajapinnan äänien soittamiseen ohjelmakoodista käsin yksinkertaisilla komennoilla.

Projektiin sisältyi useita eri ääniä kuten ääniefektejä, ambienssia ja musiikkia. Erityyppisiä ääniä tuli kohdella eri tavalla ja niiden oli oltava hallittavissa, jotta pelaaja voi valitessaan säätää äänien olemassa oloa tai voimakkuutta. Ääniefektien toteutuksessa oli myös lisävaatimuksia pelin toteutuksen kannalta kuten äänen tason, voimakkuuden, ristihäivennyksen ja vaihtamisen kannalta.

Ääniefektien toteutus projektissa vaati erilaisia ominaisuuksia äänimaailmalta. Ääniryhmiä käytettiin pelin yksiköiden reagoidessa eri pelin tapahtumiin, kuten komentojen antamiseen, milloin saatettiin käyttää eri vaihtoehtoa samantyyppisestä äänestä soittaessa. Ajoneuvojen moottoriäänissä äänen tasoa ja olemassaoloa muokattiin ajoneuvon liikkeen ja nopeuden mukaan. Pelin eri tilat koostuivat eri ambienssi äänistä sekä vaihtoehtoisista musiikkikappaleista.

Liitännäinen toteutti tarvittavat toiminnollisuudet pelin tarpeita varten hyvin, sillä pelin äänimaailma oli yksinkertainen eikä vaatinut monimutkaisia toiminnollisuuksia. Varsinkin pelin musiikin toteutus tapahtui hyvin helposti liitännäisen avulla, pieniä ongelmia lukuun ottamatta. Liitännäisen käyttö voi olla turhaa tulevilla projekteilla, varsinkin kun Unity 5-version myötä Unityn uusi äänitoteutus voi osoittautua riittäväksi.

4.4.6 VPaint

Pelimaailman luonnissa maasto on olennainen osa ympäristöä. Maaston koon ollessa yleensä suhteellisen suuri muihin pelimaailman objekteihin verrattuna, on tärkeää huomioida maaston piirtoon liittyvät kuvat, jotka voivat varata muistia merkittäviä määriä vaikuttaen suorituskykyyn. Suurien maastotekstuurien sijaan käytettiin pelissä vain muutamia toistuvia tekstuureja, joita sävytinohjelma piirsi kehittäjän määritelmän mukaisesti.

Valkyrie Entertainment-yrityksen kehittämä VPaint: Advanced Vertex Painting-liitännäisen avulla on mahdollista maalata eri tekstuureja 3D-malleille. Liitännäisessä muokataan 3D-mallin solmujen väriarvoja, minkä mukaan sävytinohjelma piirtää toistuvia tekstuureja väriarvojen mukaan. Värien muokkaamista varten liitännäiseen on luotu maalaustyökalut, missä käyttäjä voi itse työkalun avulla määrittää suoraan pelinkehitystyökalussa värejä 3D-mallille, sen sijaan että käyttäisi 3D-mallinnusohjelmaa.

Projektissa liitännäistä käytettiin pelin maastojen luomiseen. Niiden avulla voitiin luoda yksityiskohtia sisältävää maastoa säilyttäen hyvä suorituskyky sekä mahdollisuus muokata maastoa nopeasti pelinkehitystyökalussa. Huomioitavaa kuitenkin oli että kyseessä olevat 3D-mallin solmujen tiheys vaikutti maalaukseen ja siihen saatavien yksityiskoh-
tien määrään.

Unity tarjoaa omat maastotyökalut maastojen luomista varten, sisältäen maaston muodon muokkauksen, tekstuurien maalaamisen sekä puiden ja ruohon asettamisen. Projektissa ei tarvittu maaston muokkausta ja puiden asettelu tapahtui muiden 3D-mallien tapaisesti. Tämän vuoksi Unityn omat työkalut jäivät suurimmalta osalta tarpeettomiksi.

Jatkossa maaston luomiseen kannattaa harkita Unityn oman maaston käyttöä riippuen toteutettavan pelin tyylistä, esimerkiksi jos maaston muotoja olisi syytä muokata työkalun kautta. Liitännäinen tarjosi helpon käyttöliittymän sekä tavan luoda yksityiskohtia 3D-malleille säilyttäen hyvän suorituskyvyn.

4.4.7 A* Pathfinding Project

Polunetsinnän tarkoituksena on löytää mahdollisimman hyvä reitti eri pisteiden välillä. Tyypillisesti pyritään etsimään lyhintä reittiä paikasta toiseen, mutta on myös tilanteita, joissa muita tekijöitä on syytä ottaa huomioon. Polun etsintää varten luodaan graafi tai graafeja, missä etsintä suoritetaan käyttämällä polunetsintään kehitettyjä algoritmeja.

Aron Granbergin kehittämä A* Pathfinding Project-liitännäinen tarjoaa useita eri ominaisuuksia polunetsintään sekä mahdollisuuden vaikuttaa liitännäisen toteutukseen. Useiden graafien olemassaolo yhtäaikaaisesti mahdollisti erityyppisten yksiköiden liikkumisen eri graafeilla sekä yhtäaikaisen polunetsinnän eri pelitiloissa. Myös eri graafityypit ovat tuettuja liitännäisessä mukaan lukien tyypilliset verkkograafit, ruudukkograafit sekä pistegraafit. Pelitiloihin valittiin parhaiten sopiva graafityyppi. Esteiden ja alueiden määrittely oli myös hyvin tuettu liitännäisessä, minkä avulla polunetsintään voitiin vaikuttaa huomattavasti.

Projektissa polunetsintä on yksi keskeisimpiä ominaisuuksia, joita pelin eri yksiköt käyttävät tehtävätilassa sekä karttatilassa. Tehtävätilassa polunetsintä toteutettiin luomalla kaksi graafia, missä toinen oli ajoneuvoja ja toinen hahmoja varten. Näiden ominaisuudet poikkesivat hieman toisistaan, missä tavoitteena oli yksinkertaistaa eri operaatioiden logiikka. Graafien solmujen ja kaarien luomiseen vaikutti pelimaailmassa sijaitsevat eri esteet, joiden mukaan solmuja ja niiden välisiä yhteyksiä luotiin. Staattisten esteiden lisäksi oli tuettava myös dynaamisia esteitä kuten siviililiikenne sekä muut pelimaailman objektit. Alueiden määrittelyllä voitiin vaikuttaa huomattavasti polunetsintään. Teitä varten luotiin alueet, jolloin ajoneuvot suosivat niiden käyttämistä etsiessään polkua.



KUVA 8: Polunetsintä ruudukkograafi missä solmujen alueet visualisoidaan värin mukaan sekä etsityt polut piirretään kirkkaan vihreänä.

Liitännäinen mahdollistaa useita eri tapoja toteuttaa polunetsintä, milloin eri toteutuksia pystyi käyttämään pelin eri tiloihin tarpeiden mukaan. Useat eri ominaisuudet sekä mahdollisuus muokata liitännäisen toteutusta auttoivat huomattavasti projektin polunetsinnän toteuttamisessa. Unityn oma toteutus sopii hyvin projekteihin, joissa polunetsinnän tarpeet ovat suoraviivaisemmat. Liitännäisen avulla tarvittavat ominaisuudet pystyi toteuttamaan hyvin, joten vastaaviin tuleviin projekteihin liitännäinen on hyvä valinta.

4.4.8 Decal System

Siirtokuvat ovat 3D-mallin pinnalle projisoituja kuvia, joista muodostuu uusi 3D-malli. Siirtokuvien 3D-mallit voidaan yhdistää yhdeksi isoksi 3D-malliksi, jolloin kuvan sijaan käytetään tekstuuriatlasta mallin piirrosta. Näin voidaan saavuttaa vähäiset piirtokäskyt yksityiskohtien piirrosta. Siirtokuvia luodaan tyypillisesti staattisten objektien pinnalle suorituskyvyn saavuttamiseksi, mutta tekniikkaa voidaan käyttää muissakin tilanteissa tarvittaessa.

Edelweiss Interactive-yrityksen kehittämä Decal System-liitännäinen toteuttaa Unity ympäristöön siirtokuvat hyvin toteutetulla käyttöliittymällä, minkä avulla peliin oli mahdollista luoda suuria määriä yksityiskohtia käyttämättä suuria tekstuureja vain toistamalla

siirtokuva-atlaksessa olevia kuvia. Tämä osoittautui erittäin hyväksi ratkaisuksi, esimerkiksi ajoväylien kaistaviivojen luomiseen sekä muokkaamiseen myöhemmin.

Liitännäisestä on myös saatavilla pro-versio, joka tarjoaa hieman lisäominaisuuksia, jotka kuitenkin todettiin tarpeettomiksi projektin kehityksessä. Nykyään ilmainen versio ei ole enää saatavilla, joten pro-version käyttö tulee vastaan tulevilla projekteilla, jos liitännäisen tarjoamia ominaisuuksia tarvitaan.

Siirtokuvien avulla voitiin projektissa luoda yksityiskohtia pelin kenttiin vaivattomasti ilman jatkuvaa graafikoiden työtä, koska kenttäsuunnittelija pystyi helposti sijoittelemaan siirtokuvia kenttään. Unity 4 versiota käyttäessä liitännäinen on erinomainen valinta kenttien elävöittämiseen, mutta toistaiseksi liitännäistä ei ole päivitetty Unity 5 version toteutuksia varten, vaikkakin sitä on mahdollista käyttää. Siirtokuvien toteuttamiseen Unity 5 versiota varten, on mahdollista muokata Unityn laskennallisten siirtokuvien esimerkki toteutusta (Pranckevičius, A. 2015.).

4.5 Ohjelmointiympäristö

Projektiin sisältyi paljon ohjelmakoodia ja sen tuottamiseen löytyy muutamia vaihtoehtoja kehitysalustasta riippuen. Pääasiallisesti projektin tuotanto keskittyi Windows-alustalle kehittämiseen, mutta esimerkiksi siirrettäessä peliä OS X-alustalle oli käytännön syistä käytettävä kyseistä alustaa. Vartenotettavia vaihtoehtoja kehitykselle ovat Microsoftin Visual Studio, Xamarin yrityksen kehittämä MonoDevelop tai Xamarin Studio. Xamarin Studion käytöstä luovuttiin, koska se ei kyennyt eri muutosten takia jäljittämään virheitä lähdekoodista, mikä vaikeutti ohjelmakoodin toteutusta ja korjausta huomattavasti.

Pääasiallisesti projektin ohjelmakoodi toteutettiin Microsoft Visual Studio ohjelmointiympäristöllä ja mahdolliset pienet muutokset ohjelmakoodin saatettiin tehdä MonoDevelopilla OS X porttaus vaiheen aikana. Visual Studio tarjoaa huomattavan määrän pieniä ominaisuuksia, jotka nopeuttavat ohjelmakoodin luontia sekä tarjoaa mahdollisuuden käyttää ominaisuuksia lisääviä liitännäisiä, mitkä helpottavat ohjelmakoodin kirjoittamista.

Jotta Microsoftin Visual Studio ohjelmaa voisi käyttää tehokkaasti Unityn kanssa, täytyi ottaa käyttöön UnityVS-liitännäinen, mikä mahdollisti virheiden jäljittämisen lähdekoodista. Myöhemmin Microsoft hankki UnityVS-liitännäisen tekijän SyntaxTree yrityksen tehden liitännäisestä ilmaisen nimellä Visual Studio Tools for Unity (Somesegar, S. 2014).

Ohjelmointiympäristö valittiin ohjelmointitiimin harkinnan mukaisesti. Microsoft Visual Studion avulla voitiin projektiin määritellä syntaksi käytäntöjä, jolloin koodin luettavuus parani koko ohjelmointitiimin sisällä. Visual Studion liitännäiset mahdollistavat myös paljon ominaisuuksia, joiden avulla ohjelmakoodin kirjoittaminen on huomattavasti helpompaa sekä virheiden tunnistaminen nopeampaa. Tuleviin projekteihin olisi koko ohjelmointitiimin hyvä käyttää Visual Studiota sen kattavien ominaisuuksien vuoksi.

4.6 Audio toteutus

Pelin äänimaailma luotiin luvussa 4.4.5 mainitulla Unity-liitännäisellä, mikä toteuttaa tarvittavat toiminnollisuudet äänimaailman luomiseen käyttäen pohjana Unityn omaa toteutusta. Pelin äänimaailman ollessa suhteellisen yksinkertainen, sisälsi se vain kourallisen erityyppisiä ääniä.

Musiikin lisäksi pelimaailmaan kuului ääniefektejä, joita soitetaan tärkeissä pelielementeissä kuten tulipaloissa, sammutteissa sekä ajoneuvoissa. Erilaisia ambienssi ääniä luotiin pelin eri tiloille ja niiden peliympäristöille. Pelin komentoja sekä karttapelitilan tehtäviä varten luotiin myös selostusääniä.

Pelaajan oli mahdollista säätää eri äänityyppien voimakkuutta pelin asetuksista oman mieltymyksensä mukaisesti esimerkiksi säätämällä pelin musiikin kokonaan pois. Pelin äänien luonti ulkoistettiin eri tekijöille, mutta äänien toteutus peliin tapahtui pelikehitystiimin toimesta.

Äänimaailman toteutus on tyypillisesti osa-alue, johon pelinkehitystiimi keskittyy vähiten, vaikkakin äänet voivat olla suuri tekijä pelin tunnelman luonnissa. Projektissa äänimaailman toteutus aloitettiin liian myöhään tuotannossa, eikä yrityksessä ollut täysin ääniin keskitettyä työntekijää, minkä takia äänimaailman toteutus jäi heikoksi. Projektiin

luodut musiikit sen sijaan toimivat hyvin ja julkaisun jälkeen saadun palautteen perusteella pelin fanit olisivat olleet valmiita ostamaan pelin kappaleita erikseen.

4.7 Graafinen toteutus

Pelin graafinen toteutus on hyvin 3D painotteinen, missä 3D-mallit ovat tehty pelin katselukulmaa varten, minkä johdosta 3D-mallit hyvin yksinkertaisia. Graafisessa toteutuksessa keskittyy pelin tärkeimpiin elementteihin luoden pelin hahmoista sekä ajoneuvoista suhteessa muuhun maailman näyttävämpiä. Pelimaailman luomisessa käytettiin paljon toistuvia-3D malleja, mitä projektin kenttäsuunnittelija saattoi muokata nopeilla keinoilla vaihtelun luomiseksi, kuten muokkaamalla mallien kokoa, kiertoa sekä asettamalla pintakuvia mallien päälle.

Käyttöliittymää varten luotiin useita eri kuvia, minkä avulla tekstien sijaan voidaan esimerkiksi käyttää ikoneita helpomman luettavuuden saavuttamiseksi. Käyttöliittymä oli erityisen haastava toteuttaa, sillä pelaajan oli pystyttävä antamaan useita eri komentoja eri tilanteissa sekä lukemaan pelin tilannetta, minkä pohjalta pelaaja voi tehdä päätöksiä pelissä.

Ylimääräisenä resurssina graafisessa toteutuksessa toimi myös aikaisemman peliosan eri 3D-mallit, mitä saatettiin käyttää suoraan tai muokattuna eri tilanteissa kuten pelikentissä. Myös pelin henkilöstön potretti kuvien luonnissa käytettiin aikaisemman peliosan potretti kuvia.

Eri työkalujen ja aikaisempien resurssien ansiosta pelin graafinen toteutus saatiin toivotulle tasolle. Projektin henkilöstön osaaminen on iso tekijä kuinka hyvin eri menetelmillä voidaan parantaa pelin visuaalista ilmettä. Osaamisen karttumisen myötä sekä teknologian kehittyessä voidaan olettaa että tulevien projektien visuaalinen laatu paranee.

4.8 Markkinointi materiaalin luonti

Projektin kehityksen lisäksi tuli yritykselle tehtäväksi luoda materiaalia julkaisijan suorittamaa markkinointia varten kuten eri kuvankaappaukset pelistä, käsitellyt kuvat eri pelin elementeistä ja erilaiset mainosvideot. Kuvakaappauksia varten tuotannon aikana luotiin pelissä hallittava kamera, jota käyttäjä pystyi vapaasti liikuttamaan pelimaailmassa

ilman rajoituksia ja vaihtamaan kameran eri asetuksia, minkä avulla voitiin luoda pelitalanteesta poikkeavia kuvia.

Osan materiaaleista pystyi projektin henkilöstö hoitamaan tuotannon ohella, mutta varsinkin mainosvideoiden tuottaminen vaatii resursseja projektin henkilöstön ulkopuolelta hyvän lopputuloksen saavuttamiseen.

Merkittävin markkinointimateriaali oli pelin ominaisuuksia sekä pelattavuutta esittävä traileri, joka sisälsi kohtauksia pelistä sekä luotuja kohtauksia käyttäen peliin luotua materiaalia kuten hahmoja, autoja ja pelikenttiä.



KUVA 9: Kuvankaappaus pelin markkinointiin tehdystä videosta.

Markkinointi on tärkeä osa pelien menestymistä, eikä peliyritykset taikka pelikehitys tiimit osaa aina hoitaa markkinointia tavoilla, jotka edistäisivät pelin myyntiä. Markkinointia varten tuotetut materiaalit peliyrityksen sisällä olivat kelvollisia, mutta vievät turhaan resursseja pelinkehitys tiimiltä. Resurssit olisi voitu käyttää paremmin eri tuotannon tehtäviin tai markkinoinnin ulkoistamiseen. Jatkossa olisi yrityksen järkevämpää keskittyä pelien kehittämiseen ja hoitaa markkinointimateriaalin luonti eri tavoilla parhaiden tulosten saamiseksi.

5 POHDINTA

Unityn käyttöä tulevissa projekteissa tulee harkita tilannekohtaisesti, sillä loppujen lopuksi pelikehitystyökalun valinta pelin kehittämiseen riippuu täysin tilanteesta. Unityn vahvuuksiin kuuluu useiden alustojen tuki sekä erittäin joustava käyttöliittymä, mitä kehitystiimi voi laajentaa omien tarpeidensa mukaisesti. Visuaalisesti Unity jää muiden yleisimpien pelikehitystyökalujen varjoon kuten UDK sekä CryEngine, mikä ei ole suuri tekijä, jos kohde alustana on esimerkiksi mobiililaitteet. Yritys on tähän mennessä käyttänyt kaikissa projekteissaan Unityä, minkä vuoksi siirtyminen muihin pelikehitystyökaluihin voi aiheuttaa hidasteita tuotannossa uusien työmenetelmien sekä resurssien luonnissa.

Rescue 2: Everyday Heroes projektiin Unity oli selkeä valinta tuotannon laajuuden sekä aiemman version johdosta. Aikaisempien resurssien käyttö sekä aiemmissa projekteissa opitut työmenetelmät olivat olennaisia projektin loppuun viemisen suhteen. Tärkeä tekijä oli myös Unityn tarjoama Asset Store-palvelu, mistä löytyi useita valmiita ratkaisuja projektin ominaisuuksien toteuttamiseen, minkä johdosta oli mahdollista käyttää vapautuvat resurssit pelin uniikkien ominaisuuksien toteuttamiseen. Kohdealustat eivät olleet merkittävä tekijä Unityn käyttämisessä, vaikkakin siirtymä Windows- ja OS X-alustoilla oli pääsääntöisesti hyvin vaivatonta.

Projektin onnistui keskinkertaisesti. Ongelmia tuotannossa aiheuttivat pelisuunnittelun laajuus suhteessa projektin resursseihin. Tämän takia pelisuunnittelua jouduttiin muuttamaan tuotannon aikana, milloin käytetyt resurssit menivät osaksi hukkaan. Myös suuret muutokset projektin henkilöstössä aiheuttivat ongelmia, sillä uudet tiimin jäsenet pitivät perehdyttää projektiin sekä käytettäviin työkaluihin, jolloin jäi vähemmän aikaa pelin kehittämiseen. Julkaisija onneksi tuli vastaan eri asioissa, mikä mahdollisti sen, että tuotanto voitiin ongelmista huolimatta viedä loppuun saakka ja saada valmis tuote aikaiseksi, vaikkakin oli alkuperäistä suunnitelmaa myöhemmin.

Tulevissa projekteissa tulisi olla tarkkana saatavilla olevien resurssien suhteen sekä pyrkiä pitämään projektin avainhenkilöt projekteissa mukana alusta loppuun, jolloin voidaan varmistaa, että tuotanto kulkee eteenpäin vaikka muutoksia tiimin henkilöstöön tulisikin projektin aikana. Olemassa olevien resurssien käyttö on myös tärkeää ottaa huomioon,

jos niitä on mahdollista käyttää tulevissa projekteissa ilman että tuotettavien pelien laatu kärsii.

Projektinhallinnan suhteen tulisi tutkia, kuinka ketteriä menetelmiä voisi soveltaa paremmin tuleviin projekteihin, jotta pelien hauskuus ja pelattavuus voitaisiin selvittää huomattavasti aikaisemmin. Teknologia, työkalut sekä prosessit kehittyvät nopeasti, minkä takia on tärkeää löytää menetelmiä, jotta muutoksiin voitaisiin reagoida.

Unity 5 julkaistiin 3.3.2015, mikä tarjoaa useita eri päivityksiä eri osa-alueilla. Näihin kuuluvat uusi valaisujärjestelmä (Unity 2015c) ja fysiikkaan perustuva piirto (Lopez, R. 2014), joiden avulla pelien visuaalisuutta voidaan parantaa huomattavasti. Merkittävä lisä on myös Unityn uusi käyttöliittymä järjestelmä (Johansen, R. S. 2014), jonka avulla käyttöliittymien tekeminen on huomattavasti helpompaa ilman hintavia liitännäisiä. Myös useat suorituskykyä parantavat lisäykset nostavat kehitettyjen tuotteiden laatua.

LÄHTEET

Johansen R. 2014. Overview of the new UI system. Luettu 09/2015.
<http://blogs.unity3d.com/2014/05/28/overview-of-the-new-ui-system/>

Keith, C. 2010. Agile Game Development with Scrum. Addison-Wesley Professional.

Lopez, R. 2014. Physically based shading in Unity 5: A Primer. Luettu 10/2015.
<http://blogs.unity3d.com/2014/10/29/physically-based-shading-in-unity-5-a-primer/>

Pranckevičius, A. 2015. Extending Unity 5 Rendering Pipeline: Command Buffers. Luettu 11/2015.
<http://blogs.unity3d.com/2015/02/06/extending-unity-5-rendering-pipeline-command-buffers/>

Rescue 2: Everyday Heroes välietappi suunnitteludokumentti. Fragment Production Oy. Projektin sisäinen dokumentaatio

Somasegar, S. 2014. Microsoft acquires SyntaxTree, creator of UnityVS plugin for Visual Studio. Luettu 08/2015. <http://blogs.msdn.com/b/somasegar/archive/2014/07/02/microsoft-acquires-syntaxtree-creator-of-unityvs-plugin-for-visual-studio.aspx>

Steam. 2015. Rescue 2: Everyday Heroes Steamissä. Luettu 07/2015. <http://store.steam-powered.com/app/339210/?l=finnish>

Unreal Engine. 2015. Unreal Engine 4 Commercial Game Deployment Guidelines. Luettu 08/2015. <https://www.unrealengine.com/release>

Unity. 2015a. Unity 5.0. Luettu 10/2015
<https://unity3d.com/unity/whats-new/unity-5.0>

Unity. 2015b. Unity Multiplatform. Luettu 09/2015. <https://unity3d.com/unity/multiplatform>

Unity. 2015c. Unity Global Illumination. Luettu 10/2015.
<http://docs.unity3d.com/Manual/GIIntro.html>

Unity. 2015d. Get Unity. Luettu 08/2015. <https://unity3d.com/get-unity>